Part A: Java Basics (26 points)

Question A.1 (11 points)

Indicate whether each statement is TRUE or FALSE (Each question = 1 point)

(a) For int values A and B, the following condition will always be true:

```
((A > B) || (B >= A))
```

(a) T

(b) A java class can contain more than one un-parameterized constructor.

(b) F

(c) You can have more than one method called main in a class.

(c) T

(d) A stack is a data-structure that allows for the last to be the severed first.

(d) T

(e) Stacks and Queues can be implemented using an array.

(e) T

(f) Every recursion method can be transformed into an iterative one.

(f) T

(g) Static members of a class can be accessed without the need to create an instance.

(g) F

(h) Methods that are declared protected can only be called from a subclass if the keyword super is used.

(h) F

(i) The following loop will never terminate.

```
for(;;) { };
```

(i) T

(j) By assigning the value null to a reference variable the previously referenced instance will be destroyed.

(j) T

(k) The compiler detects all types of errors.

(k) F

Question A.2 (1 point)

A class P contains a private data field `num` of type `int`. C is a subclass class of P and C contains no data fields (only methods). Which one of the following statements is true?

A. There is no way for methods in C to use or change the value of `num`.

B. Methods in C can directly use the value of `num` in expressions, for example, `num=num+1`.

C. Methods in C can directly use the value of `num` in expressions such as `a = a + num`, but the value of `num` cannot be changed.

D. Methods in C can only obtain the value of `num` by means of an accessor method defined in P.

Question A.3 (1 point)

Is it possible for instances of the same class to call each other's methods?

A. No

B. Yes, but only public methods

C. Yes, but only public and protected methods

D. Yes

Question A.4 (1 point)

Which of the following statements declares a two dimensional String array?

A. `String x [10][10];`

B. `String x [10,10];`

C. `String x [][];`

D. `x = new [][];`

Question A.5 (1 point)

Float, Double and Byte have which of the following in common?

A. They represent numeric values

B. They have a primitive data type counterpart

C. They are members of the `number` package

D. All of the above

Question A.6 (1 point)

Look at the code snippet and determine the value of `z`.

```
int x = 10;
short z = x;
```

A. Undefined - The code would lead to a type error

B. 10

Question A.7 (1 point)

Which of the following is a reserved word in Java?

A. `NULL`

B. `Null`

C. `null`

Question A.8 (1 point)

Which of the following loops will not terminate?

A. `for(int I=0; I!=10; I = I+2){};`

B. `for(int I=0; I<10; I = I*2) {};`

C. `for(int I=0; true; I--)  {};`

Question A.9 (1 point)

What is method "overloading"?

A. Several methods of a class have the same name

B. Methods are redefined in a sub-class

C. Methods have many parameters.

Question A.10 (1 point)

How many constructors can be defined in a class?

A. Only one per class

B. Unlimited amount

C. Depends on the type of class

Question A.11 (1 point)

Consider the following statements:

```
int[ ] p = new int[100];
int[ ] s = p;
```

After these statements, which of the following statements will change the last element of the array p to 75?

A. `p[99] = 75;`

B. `p[100] = 75;`

C. `s[99] = 75;`

D. `s[100] = 75;`

E. Both A & C

F. Both B & D

Question A.12 (1 point)

What is the term used to describe the situation when an extended class provides a function already provided in the superclass?

A. Inheritance

B. Overriding

C. Overloading

Question A.13 (2 points)

Look at the following code snippet.

```
int x = 0;
for(int I = 0; I<12;I=I+3) { x++; };
```

What is the value of x after executing the loop?

3

Question A.14 (2 points)

Look at the following code snippet:

```
int a = 0;
int b = 1;
int c =  a++ - a + b-- + ++b;
```

0 - 1 + 1 + 1

What is the value of c?

1

Part B: Algorithm Analysis (24 points)

Question B.1 (1 point)

Let $f(n) = 7\log n + 5n^2 + \frac{1}{3}n\log n + \frac{n^3}{4}$. Which one of the following statements is true:

A. $f(n)$ is $O(n^2)$.

B. $f(n)$ is $O(n^3)$.

C. $f(n)$ is $O(\log n)$.

D. $f(n)$ is $O(n\log n)$.

Question B.2 (1 point)

Let $f(n) = 2n + 7\log n + n^2$. Which one of the following statements is true:

A. $f(n)$ is $O(n)$.

B. $f(n)$ is $O(\log n)$.

C. $f(n)$ is $O(\sqrt{n})$

D. $f(n)$ is $O(2^n)$.

Question B.3 (1 point)

True or False? In order for a function $f(n)$ to be $O(g(n))$ then $f(n)$ must be smaller than $g(n)$ for *every* non-negative integer n.

A. True

B. False

Question B.4 (1 point)

Which of the following statements about analysis of algorithms using Big-O notation is false?

A. It takes into account all possible input sizes.

B. It requires us to know the details of the hardware on which the algorithm will be implemented.

C. It can be performed by studying pseudocode rather than actual program code.

D. None of the above are false.

Question B.5 (10 points)

Determine the best case time complexity *and* the worst case time complexity of the following method. Express your final answer using Big-O notation for the worst case and Big-Ω notation for the best case, but show all of your calculations.

```
// This method sorts the integers in the array 'data' in
// ascending order.

public void sortArray(int[] data) {
   for(int i=0; i < data.length-1; i++) {
      for(int j = data.length-1; j > i; --j) {
         if( data[j] < data[j-1] ) {
            int temp = data[j];
            data[j] = data[j-1];
            data[j-1] = temp;
         }
      }
   }
}
```

4 operations

Loop in a Loop

Loop

n

n

n^2

justification

Big O → $n^2 + 4n$ $O(n^2)$

Big Ω → $n^2 + n$ $O(n^2)$

7

10

7

Question B.6 (10 points)

Determine the best case time complexity *and* the worst case time complexity of the following code. Express your answer using Big-O notation for the worst case and Big-Ω notation for the best case, but show all of your calculations.

```
// this recursive method computes x to the n-th power.

public int Exponentiation(int x, int n) {
   if( n == 0 ) return 1;
   if( n % 2 == 0 ) return Exponentiation(x*x, n/2);
   else return x * Exponentiation( x*x, (n-1)/2 );
}
```

4
10

Big O = 2^n X

Big Ω = 1 X

2

Part C: Data Structures (40 Points)

Question C.1 (1 point)

Text editors and word processors usually provide an "undo" mechanism that cancels recent editing operations and reverts to former states of a document. Which of the following would be the best data structure to support the implementation of such behavior?

A. Stack

B. Queue

C. Singly Linked List

D. Doubly Linked List

Question C.2 (1 point)

In a certain computer lab, a single printer is connected to all the computers by means of the network. At any given time, several students may be sending files to be printed, even while the printer is busy. Which of the following is the most appropriate data structure for storing print jobs while they are waiting to be printed?

A. Stack

B. Queue

C. Singly Linked List

D. Doubly Linked List

Question C.3 (1 point)

Suppose that a stack is implemented using a doubly linked list, L. Which one of the following results in a correct implementation of `push` and `pop`?

A. `push` invokes `L.addToTail(o)` and `pop` invokes `L.deleteFromHead()`.

B. `push` invokes `L.addToHead(o)` and `pop` invokes `L.deleteFromTail()`.

C. `push` invokes `L.addToHead(o)` and `pop` invokes `L.deleteFromHead()`.

D. None of the above are correct.

Question C.4 (1 point)

Suppose that a queue is implemented using a doubly linked list, L. Which one of the following results in a correct implementation of the methods `enqueue` and `dequeue`?

A. `enqueue` invokes `L.addToTail(o)` and `dequeue` invokes `L.deleteFromHead()`.

B. `enqueue` invokes `L.addToTail(o)` and `dequeue` invokes `L.deleteFromTail()`.

C. `enqueue` invokes `L.addToHead(o)` and `dequeue` invokes `L.deleteFromHead()`.

D. None of the above are correct.

Question C.5 (1 point)

Which of the following statements about the capacity of a linked list is true?

A. The capacity of a linked list must be specified when it is created.

B. The capacity of a linked list is unlimited.

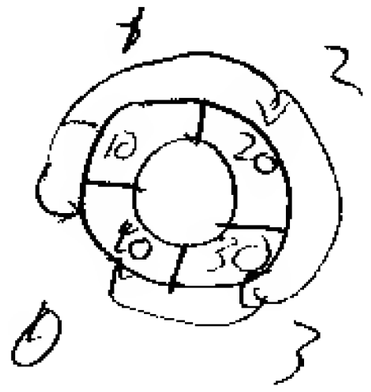
C. The capacity of a linked list limited only by the size of the physical memory of the computer.

D. None of the above are true.

Question C.6 (1 point)

Assume that a queue is implemented with a circular array of size 4. What is the value of the index variables `front` and `rear` after the following operations take place? (Assume that initially, `front` = `rear` = 0.)

```
enqueue (new Integer (10))
enqueue (new Integer (20))
enqueue (new Integer (30))
dequeue ()
enqueue (new Integer (40))
```



A. `front` = 0 and `rear` = 3.

B. `front` = 1 and `rear` = 4.

C. `front` = 1 and `rear` = 0.

D. None of the above are correct.

Question C.7 (1 point)

One difference between a queue and a stack is:

A. Queues require linked lists, but stacks do not

B. Stacks require linked lists, but queues do not

C. Queues use two ends of the structure; stacks use only one

D. Stacks use two ends of the structure, queues use only one

Question C.8 (1 point)

Which of the following correctly represents the output of the following series of stack operations: push(5), ~~push(3), pop()~~, ~~push(2)~~, ~~push(8), pop(), pop()~~, push(11), ~~push(3), pop()~~, push(7), push(5), pop(), pop(), push(4), pop(), pop().

A. 3,8,3,2,5,7,4,11

B. 3,8,2,3,5,7,4,11

C. 3,8,2,3,7,5,4,11

D. 3,8,2,3,5,7,11,4

E. None of the above

3, 8, 2, 3, 5, 7, 4, 11

4

Question C.9 (1 point)

Which of the following correctly represents the output of the following series of queue operations: enqueue(5), enqueue(6), enqueue(4), dequeue(), enqueue(4), enqueue(5), dequeue(), dequeue(), dequeue(), enqueue(12) , dequeue(), dequeue()

A. 5,6,4,5,4,12

B. 5,6,4,5,12

C. 5,6,5,4,4,12

D. 12,5,4,4,6,5

E. None of the above

Question C.10 (16 points)

Suppose you have a class `MyList` that is a **singly linked list** where the nodes are defined as follows:

```
class MyNode {
  Object element;
  MyNode next;

  MyNode(Object e) {
     this(e, null);
  }

  MyNode(Object e, Object n) {
     element = e;
     next = n;
  }
}
```

Further suppose that the class `MyList` looks like this:

```
class MyList {
  MyNode head;
  MyNode tail;

  MyList() { head = tail = null; }
  boolean isEmpty() { ...  // method code not shown }
  void addToHead(Object e) { ... }
  void addToTail(Object e) { ... }
  Object deleteFromHead() { ... }
  Object deleteFromTail() { ... }
}
```

Question continues on next page..

Question continues from previous page...

Write two new methods for `MyList`:

```
// This method should return a new list containing only elements
// that are both in the current list *and* in L.  You may
// assume that each element in a single list is unique.

public MyList intersect(MyList L) {
```

RetList = New MyList();
if (this.Element != null) if (this.Element == L.Element) × "this" is lost, not needed
{ RetList.add to tail (L.Element); this.DeleteFromHead(); } → why destroy orig. list?
else
{ Next.intersect(L); }
}
Return RetList; X

4/8

Recursion creates a new list with each recursive call.

```
}

// This method should return a new list containing only elements
// that are in *either* the current list *or* in L.  You may assume
// that each element in a single list is unique.

public MyList union(MyList L) {
```

RetList = New MyList;
if (this.Element != null)
{
if (this.Element != L.Element)
{ RetList.add to tail (L.Element); this.DeleteFromHead(); } }
else
{
Next.Union(L)
}
Return RetList;
}

Same as above?

2/4

6

Question C.11 (15 points)

Write a method that uses a stack (e.g. the one mentioned in the textbook on page 146) to determine to reverse the order of the String elements in an array.

```
public String[] reverse(String[] Data)
{
    String temp;
    for(int I =0; I < retArray[Data.length]; I++)
        {
            temp = Data[I];
            Push(temp);
        }
    for(int I =0; I < Data.length; I++)
        {
            retarray[I] = Pop();
        }
    Return retarray[];
}
```

15

15

15

Part D: Recursion (30 points)

Question D.1 (10 points)

Write a recursive method to compute the sum of the first n terms of the series

$$1 + \frac{1}{2} - \frac{1}{3} + \frac{1}{4} - \frac{1}{5} \cdots$$

```
pubic float calc(float n )
}
if (n == 0)
{return 1; }
else
{
if (n%2 == 0)
{
return (1/n - calc(n-1));
}
else
{return (1/n + calc(n-1));
}}
```

10

Question D.2 (5 points)

Determine the return value of the following recursive method when it is called with $a = \{1, 2, 3, 4, 5\}$, $b = 0$, $c = 4$ and $x = 0$:

```
public int s( int a[], int b, int c, int x ) {
    if( b>c ) return -1;
    m = (b+c)/2;
    if( a[m] == x ) return m;
    else if( x < a[m] ) { return s(a, b, m-1, x); }
    else return s(a, m+1, b, x);
}
```

Question D.3 (15 points)

Complete the method `IsPrime` below. The method should return true if `Number` is a prime number and false if it is not. An integer is prime if it can only be divided evenly by 1 and themselves.

```
public boolean IsPrime(int Number)
{
int I;
int Prime;
for (I = 0; I = Number; I++)
{
if ((Numbe % I == 0) OR (numbe % I == Number)
{Prime ++; }
}
if (Prime > 2)
{return false;}
else
{return true;}
}
```

(15)